SCHOOL OF HUMAN SCIENCES & TECHNOLOGY

RECOMMENDATION ENGINES

Professor: **MARC TORRENS ARNAL**

E-Mail: mtorrens@faculty.ie.edu

Academic Background
Ph.D., Artificial Intelligence by EPFL (École Polytechnique Fédérale de Lausanne)
Course of Enterpreneurship by The Branco Weiss Chair of Entrepreneurship and Innovation
M.S., Computer Science by the UPC (Universitat Politècnica de Catalunya)

Professional Background
CIO and Cofounder, Strands Labs., Inc. (2004 - present)
Scientific Advisor, Datumize (2014 - 2015)
Scientific Advisor, Chicisimo (2011 - present)
Postdoc Researcher, EPFL (Lausanne, Switzerland: 2002 - 2004)
CTO and Cofounder, Iconomic Systems (Lausanne, Switzerland: 1999 - 2004)

Published by IE Publishing Department.
Last revised, November 2016.

OBJECTIVES

On one hand, people are enthusiastic about the amount of information and services available on the Internet, but on the other hand, the Paradox of Choice (Less is More) indicates that people become easily overwhelmed with too many choices. How to benefit and enjoy the incredible amount of information on the Internet without being swamped into complex and unmanageable decision making processes? The answers will be given in this course!

Motivation

With the advent of the Internet the amount of information and services available to people has dramatically increased and may be overwhelming in many real life scenarios. Information Filtering in general and Recommender Systems in particular are nowadays core technologies in Internet applications and services to help users find what they need. The main idea behind a Recommendation System is to predict how a user would rate an item. Those items with high predicted ratings for a user are obviously good recommendations for that particular user. Amazon was one of the first companies introducing a Recommendation System in their applications. When a user visits a page of a book, other similar books are recommended. Netflix is another great example by recommending movies you would like based on the movies you have watched. In both examples, the goal is to help the customer to choose from a large set of options, so helping the user to decide what to buy/watch next.

Focus

The general purpose of the course it to be able to understand the main principles and challenges of a Recommendation System from different angles: business drivers, algorithms, user experience, and real-life cases. The course combines theory and practice to give the complete view necessary to be able to implement a Recommendation System in the Internet industry.

Audience and goals

The course is intended for business analysts, computer scientists, data scientists and entrepreneurs seeking to have a global view on how to make Internet applications more effective and powerful through personalization with a Recommender System. After participating in the course, you will be able to: Identify new business opportunities by incorporating recommendations technologies. Understand and implement the main recommendation algorithms. Analyse and use the different tools and methodologies involved in building a Recommendation System. Design all the software components around a recommendation engine: data collection, user experience, user interface, evaluation metrics, A/B testing, etc.

Coverage

The course includes theory with programming assignments in Python to develop basic algorithms related to Recommendation technologies.

METHODOLOGY

We will use a 10 sessions scheme to learn the basics of Recommendation Systems both from a theory and practice point of view. Different lecture methods will be used depending on the topic for each session including theory, open discussions, workshops, and readings. Participation is key to the success of the class.There will be also lab sessions in which we will learn and discuss how to actually implement the main algorithms in Python.

The sessions will be given by Marc Torrens and Ivan Tarradellas. Ivan Tarradellas is Product Manager at Strands Labs, Inc. and is leading an engineering team that develops recommendation technologies for top world-wide banks. Ivan is also teaching Lean Methodologies at the UPC BarcelonaTech.

MATERIALS

The course will be mainly supported on slides provided by the professors and paper readings that will be given prior to each session.

PROGRAM

SESSIONS 1 & 2 (FACE TO FACE)

Introduction to Recommendation Engines

- The Paradox of Choice and the Long Tail
- The Recommender Problem
- Industry Examples

Recommendation Methods

- What is a Recommendation Engine?
- Types of Recommendation Engines
    - Collaborative Filtering
    - Content-based Filtering
    - Hybrid Approaches

R.A.: Deconstructing Recommender Systems

R.A.: Data, data everywhere

These two sessions give the basics around recommendations and includes basic Python exercises to get familiar with Movielens data that will be used in the rest of the course.

SESSION 3 (FACE TO FACE)

Collaborative Filtering

- Ratings, Predictions, and Recommendations
- Explicit ratings vs Implicit ratings
- Collaborative Filtering

SESSION 4 (PRACTICE)

Recommendation Engine Lab (part 1)

- Building a Non-Personalized Recommendation Engine
- Building a Collaborative Filtering Engine

SESSION 5 (FACE TO FACE)

Content-based Filtering and Hybrid Approaches

- User Experience and Explanations
- Personalization vs Business Drivers
- Serendipity and Diversity
- Awareness and Trustiness
- Content-based Filtering
- Hybrid Approaches

SESSION 6 (PRACTICE)

Recommendation Engine Lab (part 2)

- Building a Content-based Filtering Engine
- Building an Hybrid Recommendation Engine

SESSION 7 (FACE TO FACE)

Building a Recommendation Engine in the Real World

- The Scientific Process
- Applying a Lean methodology
- Final Project Presentation
    - Introduction Milestones & Deliverables
    - Multidisciplinary Team
    - Business Canvas

SESSION 8 (PRACTICE)

Recommendation Engine Lab (part 3)

- Building your own Recommendation Engine

SESSIONS 9 & 10 (PRACTICE)

Final Project Evaluation

- Final Project Presentations
- Presentation Q&A
- Peer Evaluation

EVALUATION METHOD

Criteria	Score %
Intermediate tests	20%
Class Participation	30%
Individual work (programming)	30%
Group Presentation	20%